Are Crossings Important for Drawing Large Graphs?

Stephen G. Kobourov[1], Sergey Pupyrev[1,2], and Bahador Saket[1]

[1] Department of Computer Science, University of Arizona, Tucson, Arizona, USA
[2] Institute of Mathematics and Computer Science, Ural Federal University, Russia

Abstract. Reducing the number of edge crossings is considered one of the most important graph drawing aesthetics. While real-world graphs tend to be large and dense, most of the earlier work on evaluating the impact of edge crossings utilizes relatively small graphs that are manually generated and manipulated. We study the effect on task performance of increased edge crossings in automatically generated layouts for graphs, from different datasets, with different sizes, and with different densities. The results indicate that increasing the number of crossings negatively impacts accuracy and performance time and that impact is significant for small graphs but not significant for large graphs. We also quantitatively evaluate the impact of edge crossings on crossing angles and stress in automatically constructed graph layouts. We find a moderate correlation between minimizing stress and the minimizing the number of crossings.

1 Introduction

Graphs are often used to model a set of entities and their relationships. They are usually visualized with node-link diagrams, where vertices are depicted as points and edges as line-segments connecting the corresponding points. Many different methods for drawing graphs have been developed and they typically aim to optimize one or more *aesthetic criteria*. According to the seminal work of Purchase [20], aesthetic criteria include: number of edge crossings, number of edge bends, symmetry of the drawing, angular resolution, crossing angles, and vertex distribution. Such criteria are often proposed based on human intuition and the personal judgement of algorithm designers, and therefore the task of validating graph drawing aesthetics is of high importance.

A great deal of the prior experimental evaluations of graph drawing aesthetics utilize relatively small and nearly planar graphs and networks. For example, Purchase et al. [21] conduct a user study with graphs on 16 vertices and $18-28$ edges. Huang et al. [13,14] generate graphs having between 10 and 40 vertices. Larger graphs with 50 vertices are used by Dwyer et al. [5] but the number of edges is only 75, which results in graphs with almost tree-like structure. Real-world graphs, however, tend to be large, dense, and non-planar.

There are several of-the-shelf methods for drawing large graphs. Classical force-directed methods such as Fruchterman-Reingold [7] and Kamada-Kawai [17], and more recent multiscale variants [10, 12], define and minimize the "energy" of the layout; layouts with minimal energy tend to be aesthetically pleasing and to exhibit symmetries. Similarly, methods based on multidimensional scaling (MDS) minimize a particular energy function of the layout, called "stress" [8]. Note that the classical methods

C. Duncan and A. Symvonis (Eds.): GD 2014, LNCS 8871, pp. 234–245, 2014.

are not designed to directly optimize a specific graph drawing aesthetic criterion. Yet minimizing edge crossings remains the most cited and the most commonly used aesthetic [13,15,20,21,23]. With this in mind, we consider *the impact of edge crossings on the readability of graphs in automatically generated straight-line layouts of real-world large graphs*.

Many real-world graphs (e.g., biological networks, social networks, research citation graphs) have tens of thousands or even millions of vertices. Such graphs are not usually explored with static node-link diagrams, but rather with alternative visualization methods based on interaction, abstraction, overview-detail views, etc [1,16]. Still, static node-link diagrams with more than a hundred vertices are common today. We would like to determine a reasonable upper limit on the size of a graph, for which typical tasks can be performed using a static node-link diagram. In order to empirically define the notion of a "large graph" in this setting, we run a preliminary experiment with graphs on 100-150 vertices. For graphs with 150 vertices and density (the number of edges divided by the number of vertices) of 3.5, task accuracy is steadily below 39%, even in the most advantageous setting (e.g., high resolution display, unlimited time, the simplest path-finding tasks, graph layouts with close-to-optimal number of edge crossings, etc). The results of this preliminary experiment helped us determine useful ranges of size and density of the graphs used in our main experiment. In the main experiment, we consider *small* (40 vertices) and *large* (120 vertices) graphs. The graphs are constructed from two real-world datasets and drawn with the classical force-directed and MDS-based algorithms. We vary edge density (from 1.5 to 2.5) and the number of crossings (by a factor of two), and analyze accuracy and completion time for four tasks, frequently utilized in prior experiments. We also quantitatively evaluate the relationship between edge crossings and several other layout quality measures. Thus our contributions are two-fold:

1. We measure accuracy and completion time for four graph tasks to evaluate the effect of edge crossings on small and large graphs with varying densities. The experiments indicate that increasing the number of crossings has a negative impact, but the change is not significant for large graphs.
2. We quantitatively evaluate the impact of edge crossings on crossing angles and stress in automatically constructed graph layouts. We find a moderate correlation between minimizing stress and minimizing the number of edge crossings.

2 Related Work

Several empirical studies aim to determine the impact of various aesthetic criteria on human understanding of graph visualizations. A series of experiments by Purchase shows that many of the aesthetics are indeed important [20]. The experiments indicate that the number of edge crossings is by far the most important aesthetic, while the number of edge bends and the local symmetry displayed have a lesser impact. These results are confirmed by Huang et al. [15], who found that edge crossings significantly impact user preference and task performance. Overall, it is a common belief that minimizing the number of edge crossings is one of the most important goals in drawing graphs.

These findings have made the area of crossing minimization one of the most active research topics in the graph drawing community; see [3] for an excellent survey. However, the problem of crossing minimization is computationally hard [9], and it remains hard even when restricted to special graphs [11]. In fact, one cannot even compute in polynomial time a crossing-optimal solution for a graph obtained from a planar one by adding a single edge [4]. Given that the problem is difficult, several heuristics have been designed. The heuristics are usually hard to implement and they do not scale well with the size of a graph [3]. Hence, it is a reasonable question to ask to what extent one should try to minimize edge crossings to justify the cost.

Other graph aesthetics have also been considered. Huang et al. [14] study crossing angles (the minimum angle between pairs of crossing edges) and conclude that larger crossing angles make graphs easier to read. This motivates the research area of right-angle-crossing (RAC) drawings, where the goal is to make all crossing angles close to 90 degrees. Several studies consider the relative importance of various aesthetic criteria, which is relevant as some of them can be conflicting (e.g., minimizing crossings in planar graph drawings usually results in poor angular resolution). Huang and Huang [13] argue that the number of edge crossings is relatively more important than the crossing angles.

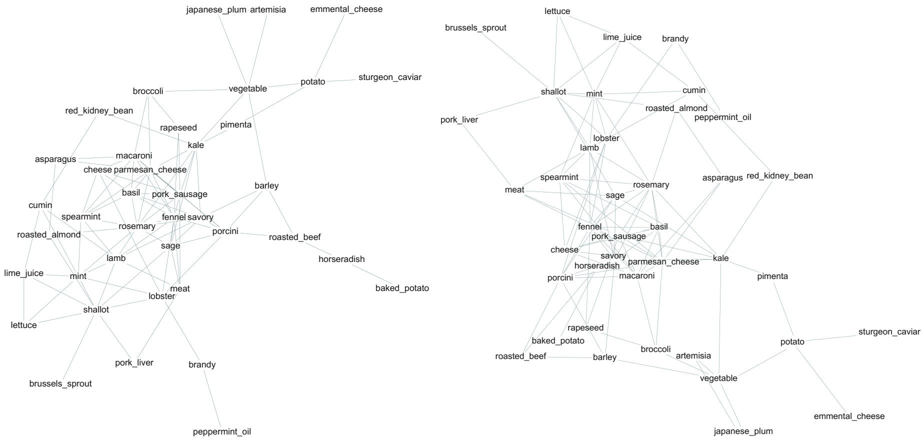
Alternative representations of large graphs and networks have also been considered. Archambault et al. [1] show that coarsening graph representations, in which several interconnected vertices are merged into metanodes, does not result in significant improvements over node-link diagrams. However, such representations might be beneficial for specific tasks in very dense graphs. Jianu et al. [16], and Saket et al. [22] investigate several methods of representing cluster information in large graphs. Their results indicate that classical node-link diagrams are not the most efficient way to visualize large clustered datasets.

3 Experiments

Objectives. We conduct a controlled experiment to explore how edge crossings affect the understandability of graph layouts. Although several studies assess the impact of crossings, a number of important questions remain open. Our specific objectives are:

1. to confirm the results of prior studies that increasing the number of edge crossings negatively impacts the usability of node-link diagrams for **small graphs**;
2. to verify whether increasing the number of edge crossings also negatively impacts the usability of node-link diagrams for **large graphs**;
3. to explore the impact of edge crossings while varying the **edge density** for both large and small graphs;
4. to analyze the impact of edge crossings on **different tasks**.

Controlled experiments in graph drawing often involve manually creating different layouts of the same graph, by varying only one aesthetic, while the others are kept unchanged. However, due to the computational hardness of the crossing minimization problem, and the use of larger graphs than those in previous studies, it is almost impossible to do this in our setting. Instead we use a different approach to accomplish a similar



(a) 139 edge crossings (b) 259 edge crossings

Fig. 1. A *small dense* graph with 40 vertices and 100 edges constructed from the Recipes dataset with (a) the *low* number of crossings and (b) the *high* number of crossings.

result by automatically generating all our drawings, without any manual postprocessing, as suggested in [13, 23]. We emphasize here that unlike most previous studies, we work only with real-world graphs and automatically computed layouts.

Our study involves a two-phase evaluation. In the first step (Experiment 1), the participant perform simple tasks on several graphs with different sizes (number of vertices) and densities (ratio of number of edges to number of vertices). This is how we determine the size of the largest graphs for which task accuracy is steadily above 50%. We use the information to design the main experiment (Experiment 2) in which we record performance, in terms of accuracy and completion time for our four tasks.

Datasets and Visualization. In order to minimize potential bias, we use two different datasets in our evaluation. The Recipes dataset contains 381 unique ingredients extracted from cooking recipes. The edges correspond to co-occurrence of the ingredients in the recipes. The GD dataset models co-authorship in the Graph Drawing conference. The vertices represent 506 authors and an edge between two vertices indicates that this pair of authors have co-authored a paper. For each dataset, we randomly sample vertices and edges creating graphs with different sizes and densities. The number of vertices is 40 (*small*) and 120 (*large*), and the edge density is 1.5 (*sparse*) and 2.5 (*dense*), making a total of 4 unweighted undirected graphs per dataset. Section 3.1 explains why we choose these sizes and densities.

We use two classical straight-line drawing algorithms implemented in GRAPHVIZ [6]. The Recipes graphs are embedded using the multidimensional scaling layout algorithm; for this purpose, we utilize the `neato` tool in GRAPHVIZ. For drawing the GD graphs, we use the force-directed placement algorithm, `fdp` in GRAPHVIZ. In order to perform our experiments, we need to have layouts of the same graph with different number of crossings. To this end, we run the layout algorithms 10,000 times on the same graph, varying the initial positions of the vertices. Since both algorithms are sensitive to the initial embedding, the resulting layouts are different. We choose two layouts of the same graph: the one with the minimum number of crossings and one with approximately twice as many crossings. These two layouts are referred to as the drawings with the *low* and *high* number of crossings; see Fig. 1. Note that neither MDS-based nor force-directed algorithms provide any guarantees about the number of crossings. However, due to the many runs for each graph, we expect that the *low* number of crossings is not too far from optimal.

Tasks. We choose the tasks for our experiments based on several considerations. First, the tasks should represent standard problems, commonly encountered when analyzing relational data. Second, the number of edge crossings in a graph visualization should likely affect task performance. Finally, the tasks should be present in existing graph task taxonomies and often utilized in other graph drawing user evaluations. With this in mind, we consider the task taxonomy for graph visualization suggested by Lee et al. [19], which categorizes the tasks into groups: topology-based, attribute-based, browsing, and overview tasks. Each of the categories specifies different subcategories. Previous studies clearly indicate that the number of edges crossings affects tasks in the topology-based category, while tasks in the other three categories are less likely to be significantly impacted by the number of crossings or do not fit in our experimental setup. The graphs in our experiments do not contain special attributes (e.g., color or shape), and hence the attribute-based tasks are not suitable. The browsing category deals with navigational tasks that do not require a specific answer, making it difficult to measure the task performance. Overview tasks are related to compound tasks (e.g., identifying changes over time, comparing the relative size of a pair of graphs) are also not suitable to our setting and less likely to be affected by the number of edge crossings. Therefore, we focus on topology-based tasks, grouped into four subcategories: connectivity, accessibility, adjacency, and common connections. For each subcategory, we choose a task that is frequently used in prior user studies on graph visualization.

Task 1: *How many edges are in a shortest path between two given nodes?*
Task 2: *What is the node with the highest degree?*
Task 3: *What nodes are all adjacent to the given node?*
Task 4: *Which of the following nodes are adjacent to both given nodes?*

The vertices for each question were randomly selected (in the case of Task 1, additionally ensuring that the pair of vertices is at most 5 edges away).

Participants and Apparatus. For the first experiment we recruited 6 participants (3 male, 3 female) aged 21–27 years (mean 23) with normal vision. For the second experiment we recruited 16 new participants (12 male, 4 female) aged 21–30 years (mean 25) with normal vision. All the participants were undergraduate and graduate science and engineering students familiar with graphs and networks. Both experiments were conducted on a computer with i7 CPU 860 @ 2.80GHz processor and 24 inch screen with 1600x900 resolution. The participants interacted with a standard mouse to complete the tasks. We used custom-built software to guide the users through the experiment by providing instructions and collecting data about time and accuracy.

3.1 Procedure: Experiment 1

Real-world graphs are typically large and non-planar. In drawings of such graphs there could be many edge crossings, which likely makes the drawings difficult to understand. To evaluate the impact of the number of crossings for different sizes and densities of graphs, while keeping the experiment to a reasonable length and complexity, we want to choose the graphs so that the average completion time is below 120 seconds and the average accuracy for a single task is higher than 50%.

To determine reasonable upper limits for the main experiment, we generated different graphs with 100-150 vertices, in increments of 10, and densities ranging from 1.5 to 3.5, in increments of 1. For every graph, we used the layout with the smallest number of crossings and for each of these layouts the participants performed the four tasks described above. The resulting completion time ranges from 63 seconds for a 100-vertex graph to 184 seconds for a 150-vertex graph. The accuracy (the number of correct answers divided by the total number of questions) ranges from 85% for 100-vertex graphs with 1.5 density to 39% for 150-vertex graphs with 3.5 density. Based on these results, we choose 120 vertices as the maximum number of vertices and 2.5 as the maximum density value for our main experiment.

3.2 Procedure: Experiment 2

An experimental system was implemented to present the 64 (2 *sizes* $\times$ 2 *number of crossings* $\times$ 2 *densities* $\times$ 2 *datasets* $\times$ 4 *tasks*) stimuli and questions for this within-subjects experiment, and to collect the participant answers and response times.

Before the controlled experiment, the participants were briefed about the purpose of the study. Although all participants were familiar with graphs, we explained all the required definitions (e.g., graphs, edges, paths). The participants then answered 8 training questions (two for each of the tasks) as quickly and as accurately as possible. The participants were encouraged to ask questions during this stage and we did not record time and accuracy for the training questions.

The main experiment consisted of the 64 tasks, presented in a reduced Latin square to counterbalance learning and order effects (to prevent participants from extrapolating new judgements from previous ones). The participants were able to zoom and pan the diagram on the screen (if needed) and were required to select one of the provided multiple choices. We recorded time and accuracy for each task. After every 12 questions, there was a break and the participants could continue when they were ready.

Hypotheses. Based on prior work and results from our preliminary experiment, we hypothesize that:

- **H1** Increasing the number of crossings negatively impacts accuracy and performance time and that impact is significant for small graphs but not significant for large graphs.
- **H2** The negative impact of increasing the number of crossings on performance is significant for both small sparse and small dense graphs.
- **H3** The negative impact of increasing the number of crossings on performance is not significant for both large sparse and large dense graphs.

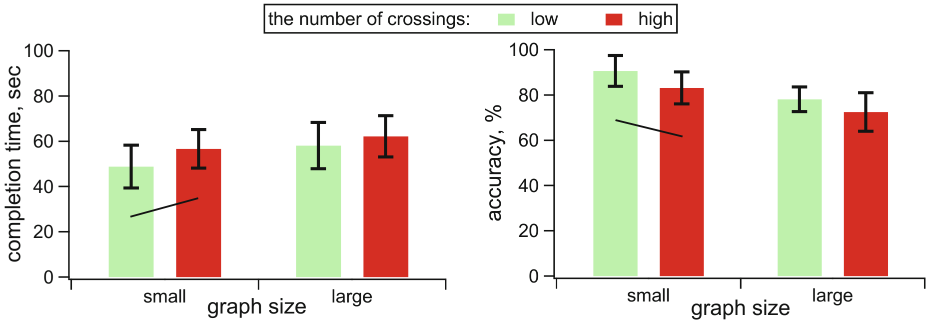


Fig. 2. Mean and standard deviation for time and accuracy in *small* and *large* graphs with different number of crossings. The differences are significant (indicated by the diagonal line segments) only for small graphs.

3.3 Results

We used a Shapiro-Wilk test to check normality of the collected data. The p-values for graphs with low/high number of crossings were 0.15 and 0.42, respectively. This, together with Q-Q plots, indicates that the data has close to normal distribution. With this in mind, we use the within-subjects t-test to analyze the results. Accuracy is measured using the number of correct trials divided by the total number of trials, thus showing a percentage. Time is measured in seconds.

Completion Time. We exclude incorrect answers, about 11% of the total, and analyze the completion time data only for the correct answers. Otherwise, the measurements of performance time might not be fair (e.g., a participant might quickly give up and give a random answer). Exclusion of incorrect answers does not decrease our sample size significantly since the average number of wrong answers per participant was 7 out of 64 questions.

Increasing the number of edge crossings for small graphs results in statistically significant reduction in performance time. For large graphs there is also a negative impact on performance time, but the results are not statistically significant; see Fig. 2. These results support H1.

Looking at the breakdown into large and small and dense and sparse provides further information. The data are summarized in Table 1, where the small (large) category refers to the average results computed for small (large) sparse and dense graphs.

Increasing the number of edge crossings results in statistically significant reduction in performance time for both small sparse and small dense graphs. This supports H2.

Increasing the number of edge crossings does not result in statistically significant reduction in performance time for large dense graphs (but the reduction is statistically significant for large sparse graphs). This partially supports H3.

Further breakdown by task, reveals more interesting results. For small graphs the main contributors to the statistically significant impacts observed earlier are Tasks 2 and 3 . For large graphs, there is a statistically significant impact for Task 1, although

Table 1. Mean (μ) and standard deviation (σ) of *Completion Time* (in seconds). Statistically significant differences between performance time in layouts with the low and high number of edge crossings are highlighted.

graphs	the number of crossings		t-test results	
	low	*high*	p-value	t-value
small	$\mu = 48.8$ $\sigma = 9.4$	$\mu = 56.6$ $\sigma = 8.4$	$p < .05$	$t(15) = 2.9$
large	$\mu = 58.0$ $\sigma = 10.1$	$\mu = 62.2$ $\sigma = 9.0$	$p = .24$	$t(15) = 2.0$
small sparse	$\mu = 44.2$ $\sigma = 11.0$	$\mu = 51.3$ $\sigma = 6.7$	$p < .05$	$t(15) = 2.4$
small dense	$\mu = 53.4$ $\sigma = 11.9$	$\mu = 62.0$ $\sigma = 11.9$	$p < .05$	$t(15) = 2.3$
large sparse	$\mu = 53.6$ $\sigma = 12.7$	$\mu = 59.8$ $\sigma = 9.6$	$p = .13$	$t(15) = 1.6$
large dense	$\mu = 62.5$ $\sigma = 11.2$	$\mu = 64.7$ $\sigma = 16.0$	$p = .61$	$t(15) = 0.5$

over all tasks the impact is not significant. Surprisingly, increasing the crossings in large graphs improved the performance time of Task 3 by 10 seconds.

Accuracy. Increasing the number of edge crossings for small graphs results in statistically significant reduction in performance accuracy. For large graphs there is also a negative impact on performance accuracy, but the results are not statistically significant; see Fig. 2. These results support H1.

Looking at the breakdown into large and small and dense and sparse provides further information; see Table 2.

Increasing the number of edge crossings results in statistically significant reduction in accuracy for small dense graphs (but the reduction is not statistically significant for small sparse graphs). This partially supports H2.

Increasing the number of edge crossings results in statistically significant reduction in accuracy for large dense graphs (but the reduction is not statistically significant for large sparse graphs). This partially supports H3.

Further breakdown by task shows that for small graphs Tasks 2 and 4 contribute to the statistically significant impacts observed earlier. Although over all tasks the impact is not significant for large graphs, there is statistically significant difference in accuracy of Tasks 1 and 2. This is counterbalanced with a statistically significant difference in accuracy in opposite direction for Task 4; see more about this below.

3.4 Discussion

Our first hypothesis (H1) is confirmed: increasing the number of edge crossings significantly affects performance time and accuracy for small graphs and the impact is not statistically significant for large graphs. The second hypothesis (H2) is partially confirmed: crossings have a statistically significant impact on time for both sparse and dense small graphs. However, the effect is not statistically significant for accuracy in both sparse and dense small graphs. The third hypothesis (H3) is also only partially confirmed: increasing the number of edge crossings has no significant impact on completion time for large graphs. However, there is statistically significant impact on accuracy for large dense graphs.

Table 2. Mean (μ) and standard deviation (σ) of *Accuracy* (in percentage). Statistically significant differences between completion time in layouts with the low and high number of edge crossings are highlighted.

graphs	the number of crossings		t-test results	
	low	*high*	p-value	t-value
small	$\mu = 94.1\%$ $\sigma = 4.3$	$\mu = 89.4\%$ $\sigma = 4.4$	$p < .05$	$t(15) = 2.8$
large	$\mu = 86.3\%$ $\sigma = 3.4$	$\mu = 83.1\%$ $\sigma = 4.0$	$p = .06$	$t(15) = 2.0$
small sparse	$\mu = 93.7\%$ $\sigma = 6.4$	$\mu = 92.9\%$ $\sigma = 6.3$	$p = .77$	$t(15) = 0.2$
small dense	$\mu = 94.5\%$ $\sigma = 7.8$	$\mu = 85.9\%$ $\sigma = 13.5$	$p < .05$	$t(15) = 2.2$
large sparse	$\mu = 89.1\%$ $\sigma = 11.1$	$\mu = 89.0\%$ $\sigma = 9.0$	$p = .81$	$t(15) = 0.2$
large dense	$\mu = 83.5\%$ $\sigma = 7.5$	$\mu = 77.3\%$ $\sigma = 13.1$	$p < .05$	$t(15) = 2.4$

It is somewhat surprising to see that increasing the crossings affects different task in markedly different ways. It is particularly unexpected to see a statistically significant positive impact on accuracy, with the increase of edge crossings, for Task 4 in large graphs! It is also worth noting that with the increase of edge crossings, the average accuracy increases for Task 3 in small graphs for Tasks 3 and 4 in large graphs. This might be due to participants paying more attention in the cases where the problem was more difficult, possibly related to the "chart junk" effect [2]. But it is also possible that edge crossings may not be as bad as we normally think, as indicated by Huang et al. [15], who found that crossings have negative effect only on some of their tasks.

There are good indications that density plays a possibly independent role, especially on accuracy. Note that we only considered two density settings (1.5 and 2.5), both of which are relatively low. Yet, together with increased number of crossings, the high density settings resulted in statistically significant decrease in accuracy both for small and large graphs. It is probably worth exploring further the nature of the interactions between size (number of vertices), density (ratio of number of edges to number of vertices) and edge crossings upper limit of density.

4 Edge Crossings and Other Aesthetic Criteria

As mentioned earlier, several traditional methods for drawing large undirected graphs are based on the assumption that minimizing a suitably-defined energy function of the graph layout results in aesthetically pleasant drawing. But do such methods also (possibly indirectly) optimize some of the standard aesthetic criteria? Next we qualitatively analyze layouts produced by `fdp` (force-directed) and `neato` (MDS-based), with respect to three commonly used and well-defined quality measures: the energy of the layout, the number of crossings, and the angles between pairs of crossing edges.

In a number of studies, the energy of a layout is defined as the variance of edge lengths in the drawing, known as *stress* [18]. Assume a graph $G = (V, E)$ is drawn with p_i being the position of vertex $i \in V$. Denote the distance between two vertices

Table 3. Correlations between three aesthetics: $r(\mathrm{En}, \mathrm{Cr})$, $r(\mathrm{En}, \mathrm{Ang})$, $r(\mathrm{Cr}, \mathrm{Ang})$ stand for the correlation coefficients r between the layout energy En, the number of crossings Cr, and the average crossing angle Ang. Absolute values between 0.7 and 1.0 indicate a strong relationship (highlighted), while absolute values between 0.3 and 0.7 indicates a moderate relationship. Negative values indicate a negative correlation.

	MDS			force-directed		
graph	$r(\mathrm{En}, \mathrm{Cr})$	$r(\mathrm{En}, \mathrm{Ang})$	$r(\mathrm{Cr}, \mathrm{Ang})$	$r(\mathrm{En}, \mathrm{Cr})$	$r(\mathrm{En}, \mathrm{Ang})$	$r(\mathrm{Cr}, \mathrm{Ang})$
GD	0.64	0.00	0.26	0.59	−0.02	−0.39
Recipes	**0.81**	−0.27	−0.15	0.61	−0.13	−0.13
Trade	**0.91**	**-0.82**	**-0.83**	0.62	0.02	−0.24
Universities	0.68	−0.53	−0.56	0.66	−0.09	−0.16
SODA	0.67	−0.69	−0.07	0.54	−0.16	0.10
IPL	**0.82**	−0.37	−0.12	**0.72**	−0.11	−0.04
TARJAN	0.62	−0.02	−0.08	0.54	−0.10	−0.04
SOCG	0.22	−0.64	−0.04	**0.72**	−0.61	−0.11
ALGO	0.41	−0.47	0.15	**0.78**	−0.64	−0.28

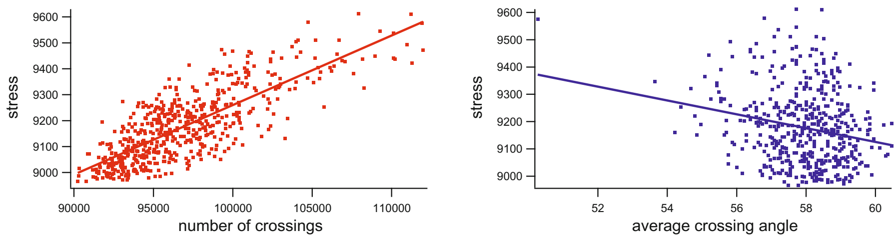
$i, j \in V$ by $||p_i - p_j||$. The energy of the graph layout is measured by

$$\sum_{i,j \in V} w_{ij}(||p_i - p_j|| - d_{ij})^2, \tag{1}$$

where d_{ij} is the ideal distance between vertices i and j, and w_{ij} is a weight factor. Typically an ideal distance d_{ij} is defined as the length of the shortest path in G between i and j. Lower stress values correspond to a better layout. We use the conventional weighting factor of $w_{ij} = \frac{1}{d_{ij}^2}$.

We run the two algorithms `fdp` and `neato` on 9 graphs for $1,000$ times on each graph. As in Section 3.2, we vary the initial layout to produce different drawings of the same graph. For each run, we measure stress, the number of edge crossings, and the average of all crossing angles of the layout. Note that Huang et al. [14] use the minimum crossing angle; in our dataset the minimum values range from 0.1 to 0.9 degrees and so the average angle provides a wider range. Then we consider the computed values for each graph as three random variables and compute the pairwise Pearson correlation coefficients; see Table 3.

The results indicate that there is a moderate positive correlation between the number of crossings and the energy of the layout for all 9 graphs processed with the force-directed algorithm and for 7 graphs processed with MDS. This means that there is a tendency for low-energy drawings to have fewer number of crossings (and vice versa). The effect is illustrated in Fig. 3, where crossings and energy are calculated for the Recipes dataset. We note here that the force-directed algorithm `fdp` (unlike `neato`) is not designed to reduce the energy function as defined by Equation (1). Yet the number of crossings is steadily correlated with the energy. This experimental evidence partially supports the observation of Dwyer et al. [5], who show that users prefer graph layouts with lower stress.



(a) $r(\mathrm{En}, \mathrm{Cr}) = 0.81$ (b) $r(\mathrm{En}, \mathrm{Ang}) = -0.27$

Fig. 3. Relationship between the energy of the drawing (stress) and (a) the number of crossings, (b) the average crossing angle. Dots represent values of the aesthetics computed for different layouts created by the multidimensional scaling algorithm for the Recipes graph.

On the other hand, there are no strong correlations between the other aesthetics. Our results indicate that the number of crossings and the crossing angles are independent in the layouts created by the two evaluated algorithms. We also note a negative correlation between the average crossing angle and the energy on 4 graphs processed with the MDS-based layout algorithm.

5 Conclusion and Future Work

All relevant materials for this study, including more detailed data analysis, are available at `http://sites.google.com/site/gdpaper2014`.

Our experimental results hopefully serve to inform designers of graph drawing algorithms that minimizing the number of edge crossings in large graphs is not as important as in small graphs. The correlation between low energy layouts and layouts with few crossings indicates that traditional energy-based methods might already result in some reduction in crossings. Although we attempted to be as diverse as possible, our results should be interpreted in the context of the specified graphs, sizes, densities, and tasks.

Due to natural limitations (e.g., length and complexity of experiments), we could not include graphs with more than 120 vertices and density greater than 2.5. Obtaining more results for larger range of the parameters would hopefully help provide a more complete picture. In our experiment we only considered relational reading of static graph drawings; results may be different in experiments that require an interpretive reading of graph drawings in the context of application domains. It would be also worthwhile to consider tasks beyond the network-topology category.

Another interesting direction would be to study in depth the effect of layout energy on understandability of graphs. Different energy function formulations (e.g., stress, distortion) likely have different impact. Evaluating such impact on a greater number of quantitatively measurable aesthetic criteria, as well as on actual tasks performance, is also a promising direction for future work.

Acknowledgements. The work supported in part by NSF grants CCF-1115971 and DEB-1053573.

References

1. Archambault, D., Purchase, C.H., Pinadu, B.: The readability of path-preserving clustering of graphs. EuroVis 29(3), 1173–1182 (2010)
2. Bateman, S., Mandryk, R.L., Gutwin, C., Genest, A., McDine, D., Brooks, C.: Useful junk? The effects of visual embellishment on comprehension and memorability of charts. In: CHI, pp. 2573–2582 (2010)
3. Buchheim, C., Chimani, M., Gutwenger, C., Jünger, M., Mutzel, P.: Crossings and planarization. In: Handbook of Graph Drawing and Visualization. CRC Press (2013)
4. Cabello, S., Mohar, B.: Adding one edge to planar graphs makes crossing number and 1-planarity hard. SIAM Journal on Computing 42(5), 1803–1829 (2013)
5. Dwyer, T., Lee, B., Fisher, D., Quinn, K.I., Isenberg, P., Robertson, G., North, C.: A comparison of user-generated and automatic graph layouts. IEEE Trans. Vis. Comput. Graphics 15(6), 961–968 (2009)
6. Ellson, J., Gansner, E.R., Koutsofios, L., North, S.C., Woodhull, G.: Graphviz - open source graph drawing tools. In: Mutzel, P., Jünger, M., Leipert, S. (eds.) GD 2001. LNCS, vol. 2265, pp. 483–484. Springer, Heidelberg (2002)
7. Fruchterman, T.M., Reingold, E.M.: Graph drawing by force-directed placement. Software: Practice and Experience 21(11), 1129–1164 (1991)
8. Gansner, E.R., Koren, Y., North, S.C.: Graph drawing by stress majorization. In: Pach, J. (ed.) GD 2004. LNCS, vol. 3383, pp. 239–250. Springer, Heidelberg (2005)
9. Garey, M.R., Johnson, D.S.: Crossing number is NP-complete. SIAM Journal on Algebraic Discrete Methods 4(3), 312–316 (1983)
10. Harel, D., Koren, Y.: A fast multi-scale method for drawing large graphs. J. Graph Algorithms Appl. 6(3), 179–202 (2002)
11. Hliněný, P.: Crossing number is hard for cubic graphs. J. Comb. Theory B 96(4), 455–471 (2006)
12. Hu, Y.: Efficient, high-quality force-directed graph drawing. Mathematica Journal 10(1), 37–71 (2005)
13. Huang, W., Huang, M.: Exploring the relative importance of number of edge crossings and size of crossing angles: A quantitative perspective. Advanced Intelligence 3(1), 25–42 (2014)
14. Huang, W., Eades, P., Hong, S.H.: Larger crossing angles make graphs easier to read. Visual Languages & Computing 1 (2014)
15. Huang, W., Hong, S.H., Eades, P.: Layout effects on sociogram perception. In: Healy, P., Nikolov, N.S. (eds.) GD 2005. LNCS, vol. 3843, pp. 262–273. Springer, Heidelberg (2006)
16. Jianu, R., Rusu, A., Hu, Y., Taggart, D.: How to display group information on node–link diagrams: an evaluation. IEEE Trans. Vis. Comput. Graphics (to appear, 2014)
17. Kamada, T., Kawai, S.: An algorithm for drawing general undirected graphs. Inf. Proc. Let. 31(1), 7–15 (1989)
18. Koren, Y., Çivril, A.: The binary stress model for graph drawing. In: Tollis, I.G., Patrignani, M. (eds.) GD 2008. LNCS, vol. 5417, pp. 193–205. Springer, Heidelberg (2009)
19. Lee, B., Plaisant, C., Parr, C., Fekete, J.D., Henry, N.: Task taxonomy for graph visualization. In: BELIV, pp. 81–85. ACM Press (2006)
20. Purchase, H.C.: Which aesthetic has the greatest effect on human understanding? In: DiBattista, G. (ed.) GD 1997. LNCS, vol. 1353, pp. 248–261. Springer, Heidelberg (1997)
21. Purchase, H., Cohen, R., James, M.: Validating graph drawing aesthetics. In: Brandenburg, F.J. (ed.) GD 1995. LNCS, vol. 1027, pp. 435–446. Springer, Heidelberg (1996)
22. Saket, B., Simonetto, P., Kobourov, S., Börner, K.: Node, node-link, and node-link-group diagrams: An evaluation. In: IEEE InfoVis (to appear, 2014)
23. Ware, C., Purchase, H.C., Colpoys, L., McGill, M.: Cognitive measurements of graph aesthetics. Information Visualization 1(2), 103–110 (2002)